

## 1 Forelesning 6

### 1.1 Divide & Conquer

- En nyttig strategi når man skal finne effektive algoritmer.
- Divide
  - Del opp input og løs hver del for seg
- Conquer
  - Finn en måte å slå sammen de to delene til løsning for hele input

### 1.2 Mergesort

- Divide
  - Lag to lister som hver har halvparten av elementene
  - Sorter de to små listene
- Conquer
  - Gitt to små sorterte lister, slå disse sammen til en stor sortert liste.
- La oss implementere mergesort.

### 1.3 Nedre grense på kjøretid

- Vi kan se å sortering som å velge den rette ordningen blant alle mulige ordninger av en liste.
- Hvor mange måter er det å ordne  $n$  elementer? ( $n!$ )
- Tenk deg en liste av alle mulige ordninger
- Gjør en sammenligning av 2 elementer  $a$  og  $b$ .
- Blant alle ordninger i listen kan vi
  - Enten Krysse ut de der  $a < b$
  - Eller krysse ut de der  $b < a$
- Hvor mange ganger må vi dele på 2 før  $n!$  blir 1?
- $\log(n!)$

### 1.4 Sorter $n$ heltall

- Hvor for kan vi sortere  $n$  heltall der alle tallene er
  - Mellom 0 og 1000
  - Mellom 0 og  $n$
  - Mellom 0 og  $n^2$

## 1.5 Bucket Sort

- Ikke egentlig en sorteringsalgoritme
- Deler opp i grupper (buckets/bins)
  - For alle par  $a, b$  har vi:  $a \leq b \leftrightarrow \text{bucket}(a) \leq \text{bucket}(b)$
- Trenger funksjon for bucket nummer
  - Eksempel: String, 26 buckets,  $a$  i første,  $b$  i andre.
- Radix sort ligner på bucket sort
  - Deler igjen opp hver bucket basert på 2. bokstav.